

WHAT IS CLAIMED IS:

SUB A17

1. A method of indexing occurrences of a value in at least one data record using a bit vector representation, the method comprising:

5 associating a bit vector representation with a value;

associating a bit position of the bit vector representation to the at least one record;

10 determining whether the value exists in the at least one data record; and

assigning a value to the bit position in the bit vector representation based on the outcome of the determining step;

15 synchronizing the bit position with the value to reflect any updates to the value.

2. A method according to Claim 1, further comprising:

20 encoding the bit vector representation.

3. A method according to Claim 2, wherein the bit vector representation comprises a sequence of bits, encoding the bit vector representation further comprising:

25 determining whether a frequency of a binary digit is less than a threshold value; and

001220 "STEEL" 360

storing as the encoded bit vector representation at least one position of the binary digit in the bit vector representation.

5 4. A method according to Claim 3, wherein the threshold is the number of bits used to store a number.

10 5. A method according to Claim 2, wherein the bit vector representation comprises a sequence of binary digits, encoding the bit vector representation further comprising:

15 determining whether a size of a region of like binary digits is greater than a threshold value; and

storing as the encoded bit vector representation a representation of the region.

20 6. A method according to claim 5, wherein the representation comprises a start and end designation pair that represents the start and ending bits of the region.

25 7. A method according to Claim 5, wherein the threshold is twice the number of bits used to store a number.

0044346"004400
0044346"004400

8. A method according to Claim 1, further comprising:

compressing the encoded bit vector representation.

5

9. A method according to Claim 1, wherein the bit vector representation is compressed using a compression technique.

10

10. A method according to Claim 1, wherein the bit vector representation is encoded and compressed.

15

11. A method according to Claim 1, wherein the data structure is a record in a database.

20

12. A method according to Claim 1, further comprising:

examining the bit vector representation to determine whether the data record contains the value.

25

13. A method according to Claim 1, wherein plural bit vector representations exist each representing a discrete value, the method further comprising:

004346-08400
007280-94960

determining whether the data record contains more than one of the values by performing a bit-level operation on the corresponding bit vector representations.

5

14. A method according to Claim 9, wherein the bit-level operation is an "OR" operation.

10

15. A method according to Claim 13, wherein the operation is an "AND" operation.

15

16. A method of identifying combinations of values used in at least one data record comprising fields for storing the values, the method comprising:

20

creating a first bit vector representation for a first value, the first bit vector representation identifying use of the first value in the at least one data record;

creating a second bit vector representation for a second value, the second bit vector representation identifying use of the second value in the at least one data record; and

25

performing a bit-level operation on the first and second bit vector representations.

20250726

18. A method according to Claim 17,
wherein the "AND" operation is a bit-wise "AND"
returning a bit corresponding to each of the at
least one data record identifying whether a
combination of the first and second values exist in
the at least one data record.

19. A method according to Claim 17, wherein the "AND" operation is a logical "AND" returning a single result representing whether any of the at least one data record contains a combination of the first and second values.

20. A method according to Claim 16,
further comprising:

updating the at least one data record.

21. A method according to Claim 20,
wherein the update to the at least one data record
changes the first value, the method further
comprising:

updating the first bit vector
representation to reflect the update to the at least
one data record.

22. A computer-readable memory medium in which computer-executable process steps are stored, the process steps for indexing occurrences of a value in at least one data record using a bit vector representation, wherein the process steps comprise:

an associating step to associate a bit vector representation with a value;

an associating step to associate a bit position of the bit vector representation to the at least one record;

a determining step to determine whether the value exists in the at least one data record; and

an assigning step to assign a value to the bit position in the bit vector representation based on the outcome of the determining step;

a synchronizing step to synchronize the bit position with the value to reflect any updates to the value.

23. A computer-readable memory medium according to Claim 22, further comprising:

an encoding step to encode the bit vector representation.

24. A computer-readable memory medium according to Claim 23, wherein the bit vector representation comprises a sequence of bits,

encoding the bit vector representation further comprising:

a determining step to determine whether a frequency of a binary digit is less than a threshold value; and

a storing step to store as the encoded bit vector representation at least one position of the binary digit in the bit vector representation.

25. A computer-readable memory medium according to Claim 24, wherein the threshold is the number of bits used to store a number.

26. A computer-readable memory medium according to Claim 23, wherein the bit vector representation comprises a sequence of binary digits, encoding the bit vector representation further comprising:

a determining step to determine whether a size of a region of like binary digits is greater than a threshold value; and

a storing step to store as the encoded bit vector representation a representation of the region.

27. A computer-readable memory medium according to claim 26, wherein the representation

004250" 972249500

28. A computer-readable memory medium according to Claim 26, wherein the threshold is twice the number of bits used to store a number.

a compressing step to compress the encoded bit vector representation.

30. A computer-readable memory medium according to Claim 22, wherein the bit vector representation is compressed using a compression technique.

31. A computer-readable memory medium according to Claim 22, wherein the bit vector representation is encoded and compressed.

32. A computer-readable memory medium according to Claim 22, wherein the data structure is a record in a database.

33. A computer-readable memory medium according to Claim 22, further comprising:

an examining step to examine the bit vector representation to determine whether the data record contains the value.

5 34. A computer-readable memory medium according to Claim 22, wherein plural bit vector representations exist each representing a discrete value, the method further comprising:

10 a determining step to determine whether the data record contains more than one of the values by performing a bit-level operation on the corresponding bit vector representations.

15 35. A computer-readable memory medium according to Claim 30, wherein the bit-level operation is an "OR" operation.

20 36. A computer-readable memory medium according to Claim 34, wherein the operation is an "AND" operation.

25 37. A computer-readable memory medium in which computer-executable process steps are stored, the process steps for identifying combinations of values used in at least one data record comprising fields for storing the values, wherein the process steps comprise:

004316-00100
001280-91E4960

5 a creating step to create a second bit
vector representation for a second value, the second
bit vector representation identifying use of the
second value in the at least one data record; and

10

15

20

25

41. A computer-readable memory medium according to Claim 37, further comprising:

an updating step to update the at least one data record.

5

42. A computer-readable memory medium according to Claim 41, further comprising:

a determining step to determine whether the update to the at least one data record effects the first bit vector representation;

10

an updating step to update the first bit vector representation, if it is determined that the update to the at least one data record effects the first bit vector representation;

15

a determining step to determine whether the update to the at least one data record effects the second bit vector representation; and

20

an updating step to update the second bit vector representation, if it is determined that the update to the at least one data record effects the second bit vector representation.

25

43. Computer-executable process steps stored on a computer readable medium, said computer-executable process steps for indexing occurrences of a value in at least one data record using a bit vector representation, said computer-executable process steps comprising:

004230" 9Feb96

code to associate a bit vector representation with a value;

code to associate a bit position of the bit vector representation to the at least one record;

code to determine whether the value exists in the at least one data record; and

code to assign a value to the bit position in the bit vector representation based on the outcome of the determining step;

code to synchronize the bit position with the value to reflect any updates to the value.

44. A computer-executable process steps according to Claim 43, further comprising:

code to encode the bit vector representation.

45. A computer-executable process steps according to Claim 44, wherein the bit vector representation comprises a sequence of bits, encoding the bit vector representation further comprising:

code to determine whether a frequency of a binary digit is less than a threshold value; and

code to store as the encoded bit vector representation at least one position of the binary digit in the bit vector representation.

0044346 0044346

46. A computer-executable process according to Claim 45, wherein the threshold number of bits used to store a number.

47. A computer-executable process according to Claim 44, wherein the bit vector representation comprises a sequence of binary digits, encoding the bit vector representation further comprising:

code to determine whether a size of region of like binary digits is greater than threshold value; and

code to store as the encoded bit vector representation a representation of the region

48. A computer-executable process according to claim 50, wherein the representation comprises a start and end designation pair that represents the start and ending bits of the region

49. A computer-executable process according to Claim 47, wherein the threshold is twice the number of bits used to store a number

50. A computer-executable process according to Claim 43, further comprising:

code to compress the encoded bit vector representation.

5 47. A computer-executable process steps
according to Claim 44, wherein the bit vector
representation comprises a sequence of binary
digits, encoding the bit vector representation
further comprising:

```
10         code to determine whether a size of a
        region of like binary digits is greater than a
        threshold value; and
```

```

code to store as the encoded bit vector
representation a representation of the region.

```

48. A computer-executable process steps according to claim 50, wherein the representation comprises a start and end designation pair that represents the start and ending bits of the region.

49. A computer-executable process steps according to Claim 47, wherein the threshold is twice the number of bits used to store a number.

25 50. A computer-executable process steps
according to Claim 43, further comprising:

code to compress the encoded bit vector
representation.

5

10

15

20

25

code to determine whether the data record contains more than one of the values by performing a bit-level operation on the corresponding bit vector representations.

56. A computer-executable process steps according to Claim 51, wherein the bit-level operation is an "OR" operation.

5 57. A computer-executable process steps according to Claim 55, wherein the operation is an "AND" operation.

10 58. Computer-executable process steps stored on a computer readable medium, said computer-executable process steps for identifying combinations of values used in at least one data record comprising fields for storing the values, said computer-executable process steps comprising:

15 code to create a first bit vector representation for a first value, the first bit vector representation identifying use of the first value in the at least one data record;

20 code to create a second bit vector representation for a second value, the second bit vector representation identifying use of the second value in the at least one data record; and

code to perform a bit-level operation on the first and second bit vector representations.

25 59. A computer-executable process steps according to Claim 58, wherein the bit-level operation is an "AND" operation.

001280" 912E11950

60. A computer-executable process steps according to Claim 59, wherein the "AND" operation is a logical "AND" returning a bit corresponding to each of the at least one data record identifying whether a combination of the first and second values exist in the at least one data record.

61. A computer-executable process steps according to Claim 59, wherein the "AND" operation is a bit-wise "AND" returning a single result representing whether any of the at least one data record contains a combination of the first and second values.

62. A computer-executable process steps according to Claim 58, further comprising:
code to update the at least one data record.

63. A computer-executable process steps according to Claim 62, further comprising:

code to determine whether the update to the at least one data record effects the first bit vector representation;

code to update the first bit vector representation, if it is determined that the update to the at least one data record effects the first bit vector representation;

004330" 3434300

code to determine whether the update to
the at least one data record effects the second bit
vector representation; and

5 code to update the second bit vector
representation, if it is determined that the update
to the at least one data record effects the second
bit vector representation.

ADD B27

00000000000000000000000000000000